

SPRING BOOT

Project Requirement

DEPENDENCY

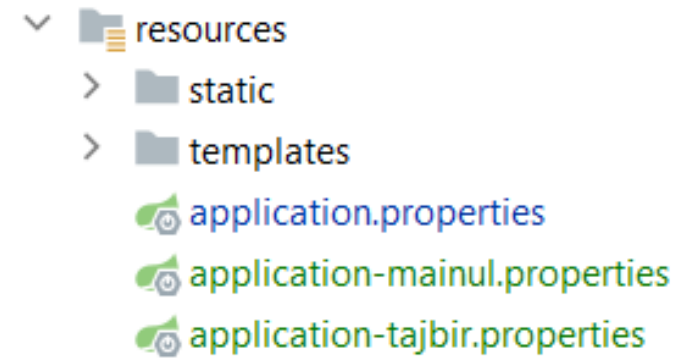
- Spring Boot DevTools
- Lombok
- Spring Web
- Thymeleaf
- Spring Data JPA
- MySQL Drive
- PostgreSQL Drive

PROFILING

- ❑ application.properties
- ❑ application-dev.properties
- ❑ application-test.properties

➤ [Code Link](#)

Push in Git : Profiling



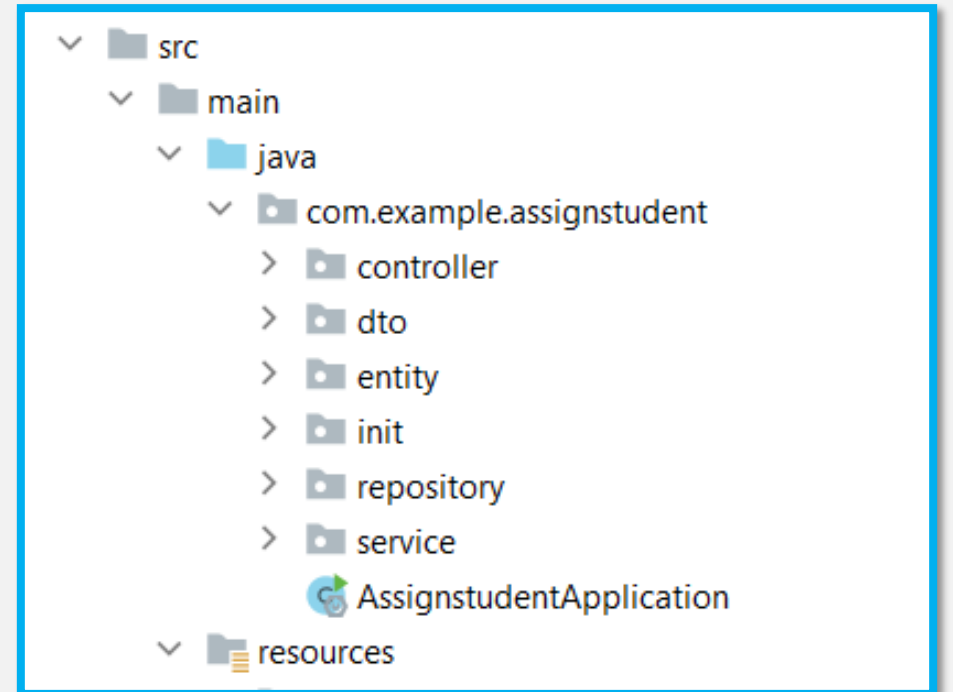
A screenshot of a file explorer window with a blue border. It shows a tree view of a project's resources. The 'resources' folder is expanded, showing subfolders 'static' and 'templates', and three property files: 'application.properties' (in blue), 'application-mainul.properties' (in green), and 'application-tajbir.properties' (in green). Each file has a small green icon with a power symbol to its left.

```
resources
├── static
├── templates
├── application.properties
├── application-mainul.properties
└── application-tajbir.properties
```

CREATE DIRECTORIES

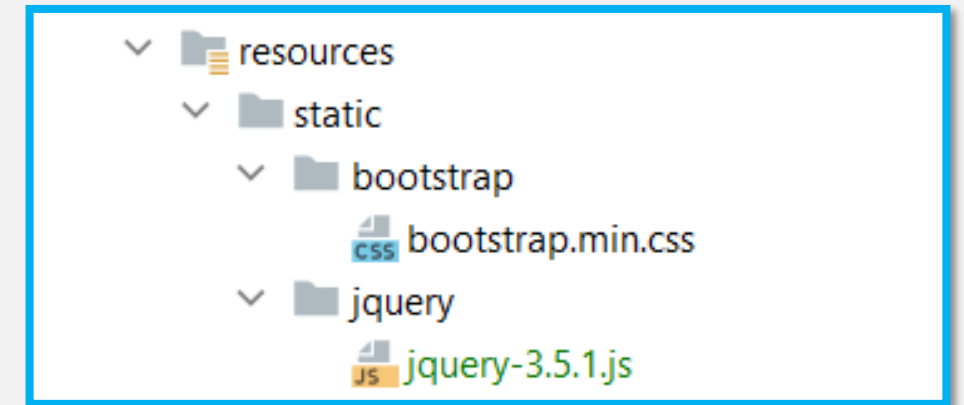
1. entity
2. dto
3. repository
4. controller
5. service
6. init
7. exception

Push in Git : Directory Structure



ADD BOOTSTRAP & JQUERY

1. Bootstrap [Download link](#)
2. jQuery [Download link](#)



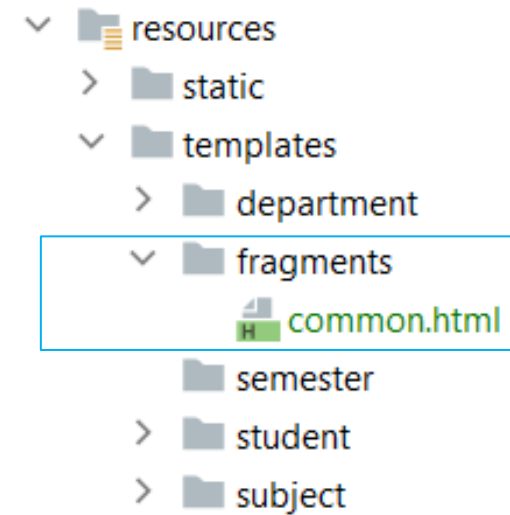
Push in Git :Add Bootstrap & jQuery Library

TEMPLATING

- ❑ Templating View Page Common Html such as Navigation Bar, Head Section.

➤ [Code Link](#)

Push in Git :Templating With Navbar & Head Section

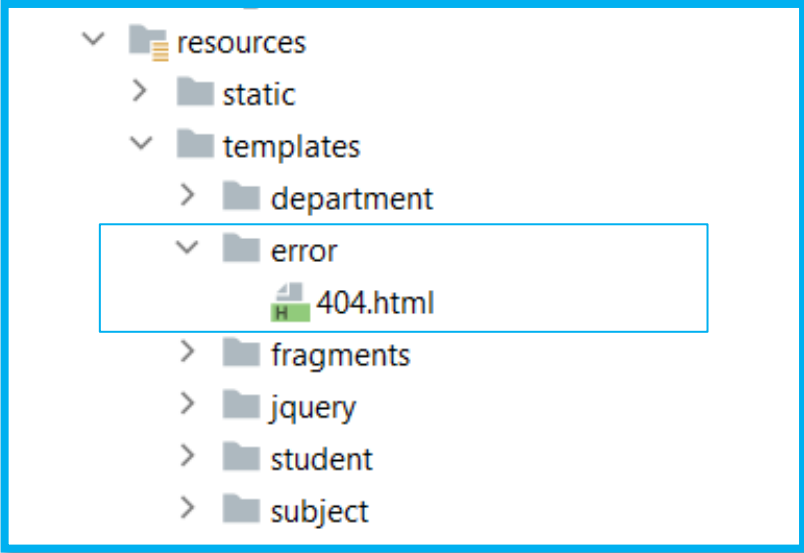


EXCEPTION CODE

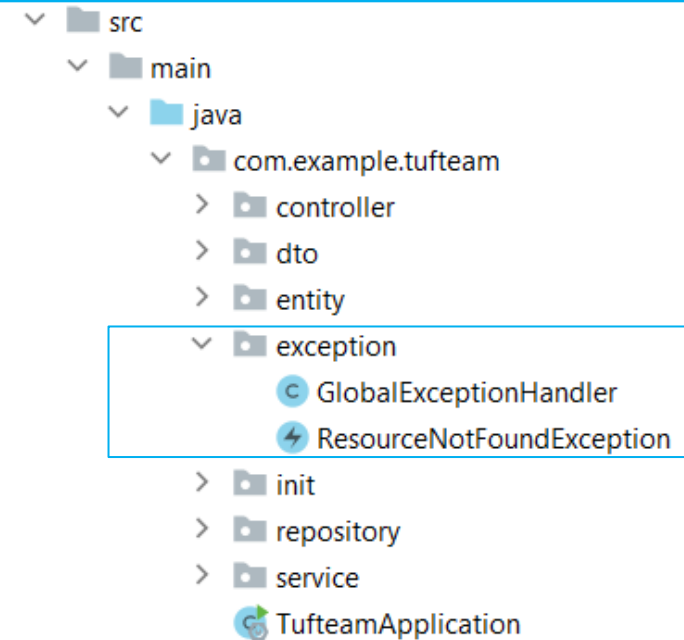
- ❑ Create 404 error View Page
- ❑ Create GlobalExceptionHandler.java class
- ❑ Create ResourceNotFoundException.java class

➤ [Code Link](#)

Push in Git : Exception Code



```
resources
├── static
├── templates
│   ├── department
│   └── error (highlighted)
│       └── 404.html
├── fragments
├── jquery
├── student
└── subject
```



```
src
├── main
│   └── java
│       ├── com.example.tufteam
│       │   ├── controller
│       │   ├── dto
│       │   ├── entity
│       │   └── exception (highlighted)
│       │       ├── GlobalExceptionHandler
│       │       └── ResourceNotFoundException
│       ├── init
│       ├── repository
│       └── service
└── TufteamApplication
```

STORIES DETAILS

STORY I

☐ Save Student

Push in Git : Save New Student By Traditional way

- Save Student View Page

Student Name

Age

Gender (Radio Button)

Save

Save Process in Traditional approach

STORY 2

☐ Save Department

Push in Git : Save Department by Traditional Way

- Save Department View Page

Department Name

Code

Save

Save Process in Traditional approach

STORY 3

- ❑ Assign Students in Department
 - ✓ Save By Department Entity
 - ✓ Multiple Students Can be Assign
 - ✓ Assigned Student can't assign again & not show in student dropdown list.
- Add relations in both Student & Department entities.

Push in Git : Git Commit- by Relation names

- Assign Student in Department View Page

The diagram illustrates the 'Assign Student in Department View Page' interface. It features a light gray background with a white rectangular area containing the following elements:

- A dropdown menu labeled 'Department List' with a downward arrow icon.
- A second dropdown menu labeled 'Student List' with a downward arrow icon.
- A blue button labeled 'Save'.
- A text box at the bottom labeled 'Save Process in Traditional approach'.

STORY 4

☐ All Student

- ✓ Get All Student By Student Entity
- ✓ Non-Assign Student Department Shows 'Not Assign Yet'
- ✓ Add Update Delete Action

Push in Git : Show All Student With Edit & Delete Action

- Show All Student View Page

All Student

ID	Name	Age	Gender	Department	Action
01	Tajbir	26	Male	CSE	Edit Delete
02	Nuva	20	Female	Not Assign Yet	Edit Delete

Show Process in Traditional approach